

Direct simulation of the infinitesimal dynamics of semi-discrete approximations for convection-diffusion-reaction problems

Flavius Guiaş

Preprint 2009-04

März 2009

Direct simulation of the infinitesimal dynamics of semi-discrete approximations for convection-diffusion-reaction problems

Flavius Guiaş

*Faculty of Mathematics, Dortmund University of Technology, Vogelpothsweg 87, 44221
Dortmund, Germany.*

Abstract

In this paper a scheme for approximating solutions of convection-diffusion-reaction equations by Markov jump processes is studied. The general principle of the method of lines reduces evolution partial differential equations to semi-discrete approximations consisting of systems of ordinary differential equations. Our approach is to use for this resulting system a stochastic scheme which is essentially a direct simulation of the corresponding infinitesimal dynamics. This implies automatically the time adaptivity and, in one space dimension, stable approximations of diffusion operators on non-uniform grids and the possibility of using moving cells for the transport part, all within the framework of an explicit method. We present several results in one space dimension including free boundary problems, but the general algorithm is simple, flexible and on uniform grids it can be formulated for general evolution partial differential equations in arbitrary space dimensions.

Key words: stochastic simulation; Markov jump processes; diffusion; convection; reaction; moving cells

2000 MSC: 65C20, 68U20, 65M06, 65M20, 65M99, 60J75

PACS: 82.20.Wt, 87.10.Mn

Email address: `flavius.guias@math.tu-dortmund.de` (Flavius Guiaş)

1. Introduction

Many complex processes in natural sciences require stochastic models for their simulation. The principle of rule-based modeling is to specify the microscopic dynamics of the system and to simulate them step by step, the cumulated result being an approximation of the macroscopic limiting dynamics. A typical example in this sense is given by coagulation processes (see [1] for a comprehensive review). In the simplest form, particles with size parameters x and y coalesce at rate $K(x, y)$ to form a particle with size parameter $x + y$. The macroscopic dynamics of this system is given by the *Smoluchowski equation*, which is an integral equation if the range of the parameters is continuous and an infinite system of ordinary differential equations if the parameters are taken as integers.

In a spatially inhomogeneous setting multi-compartment models are frequently employed. One considers homogeneous reactors where the infinitesimal dynamics are governed by a rule-based model and, in addition to this, exchange of reactants between compartments takes place due to transport or diffusion. Examples involving one reactant driven by diffusion and nonlinear reaction are analyzed e.g. in [2] and [3]. In [8] a coagulation-diffusion model based on the same principles is considered. A spatial domain is discretized by a uniform grid and each grid point (cell) is considered as a homogeneous reactor. The diffusion is modeled as a random walk of the particles on the underlying grid. Theoretical convergence results were established, but the approach turned out to be inadequate from the numerical point of view. Further steps towards the numerical simulation of coagulation-diffusion processes were done in [9] and [10], where diffusion was simulated by a stochastic (mass conservative) flux-method and the coagulation dynamics by a mass flow algorithm.

The attempt of coupling the Monte Carlo method for the reaction kinetics with a usual deterministic scheme for the motion part of the dynamics has to face the problem that the time steps of the latter are not adapted to the possibly fast time scale of the reactions. Within the relatively large time intervals of the deterministic scheme the possibility of significant changes in the spectrum of reactants exists (for example the appearance of large clusters in coagulation, especially in the gelation regime) and of which the deterministic scheme is not aware.

For an overview of standard deterministic methods for convection-diffusion-reaction problems the reader may consult the monograph [12]. Monte Carlo

methods are usually employed only in special situations, for example if the particles in a given spatial location have attached a velocity parameter which varies in a continuous range. Transport occurs then due to this velocity which in turn is subject to shocks, the discontinuous changes being modeled by a linear integral term. The solution of this type of equations is defined therefore on a higher-dimensional space, in a usual setting we have three spatial variables and three components of the velocity. For details see [14], with the mention that transport and diffusion are treated separately and the latter is approximated by numerical schemes for stochastic differential equations.

In order to couple the available stochastic kinetic methods for homogeneous reactors with spatial motion, one has to use a scheme which completely keeps track of the transitions on a microscopic level which correspond to all components of the dynamics: convection, diffusion or reaction. While convection and reaction can be simulated by jump processes in a straightforward way, the same straightforward approach for simulating diffusion (using random walks) turns out to be computationally expensive and inaccurate. In this paper we use therefore the approach of direct simulation of the dynamics of the finite-dimensional ODE-system given by the method of lines. We are mainly interested in the one-dimensional problem

$$\partial_t u + v(u)u_x = (f(u)u_x)_x + R(u). \quad (1)$$

All examples considered here are special cases of this form, but introducing a spatial dependence of the velocity and of the diffusion coefficient yields no difficulties.

The present material is organized as follows. In Section 2 we describe the general algorithm for approximating systems of ordinary differential equations by Markov jump processes. These systems can be for example semi-discrete approximations of certain partial differential equations. In contrast to the deterministic explicit schemes, the feature of this method is that the increment by which the components of the solution are changed is fixed, while the time steps are variable. For a general multidimensional system of the form $\dot{x} = F(x)$ the probability of selecting the i -th component for the next modification of magnitude $1/N$ is proportional with $|F_i(x)|$ while the time step is exponentially distributed with parameter $N \sum |F_i(x)|$. Its expected value is therefore $(N \sum |F_i(x)|)^{-1}$. Note that in a transition step we change only one component of the state of the process. This general pattern we call the direct ODE simulation method. In the same section a theoretical result concerning the fluctuations of the method is also given.

In Section 3 we compare exemplarily three stochastic approaches of simulating linear diffusion on the line: the random walk method, the flux simulation and the direct ODE simulation method. We consider a uniform grid on $[0, 1]$ with size h . For the random walk method we consider $O(N)$ particles with weights $1/N$ distributed on the grid. Each element of a pair of neighbouring grid points may be source or target of a particle jump, the corresponding concentration being reduced by $1/N$ respectively increased by $1/N$. For the other component the change has the opposite sign. The flux method however allows only directed jumps from higher to lower concentration. Both methods are mass conservative, since mass can be lost or gained only at the boundary as specified by the boundary conditions. The direct ODE simulation method changes in a single step the value of the density only in one grid point instead of two, according to the scheme presented in Section 2. The total mass is therefore conserved only in the limit for large N . The magnitude of the change in every simulation step is of $1/N$ for all three methods (this quantity being regarded as a numerical weight) but the corresponding rates are $O(Nh^{-2})$, $O(Nh^{-1})$ and $O(N)$ respectively for the random walk, flux simulation and direct ODE simulation method. This has as a consequence the fact that the first method is by a factor of h^{-1} slower than the second, which in turn is by the same factor slower than the third. Moreover, we point out theoretically as well as numerically the fact that the fluctuations induced by the three stochastic methods are in about the same proportion as their speed. This shows the enormous advantage of the direct ODE simulation method on the other two, in speed as well as in precision.

In Section 4 we consider reaction-diffusion problems which are standard examples for employing deterministic schemes with grid adaptivity in time and space. In our examples the time adaptivity is given intrinsically by the stochastic method and the grid is considered to be uniform. In Section 5 we simulate diffusion on non-uniform grids by the same basic principle of an explicit stochastic scheme, a fact that is fairly impossible in explicit deterministic methods. This feature opens the possibility of using moving grids. In Section 6 we consider cells which are transported according to the corresponding velocity and diffusion is simulated on a non-uniform grid configuration. We apply this method to the viscous Burgers' equation. The use of moving grid techniques as presented in [6] for reaction-diffusion equations with solutions which develop traveling fronts (as the examples presented in Section 4) is also possible, but is not considered here. Section 7 deals with an example of a free boundary problem, namely the Black-Scholes equation

for American put options. The motion of the free boundary emerges naturally by simulating the infinitesimal dynamics of the system, a feature which shows the flexibility of the method. Finally, in Section 8 we conclude with remarks concerning the features of the approach presented here and possible improvements and generalizations. Of interest would be the extension of the method in higher space dimensions in the case of non-uniform grids.

2. General description of the direct ODE simulation method

Consider a scalar differential equation of the form $\dot{x} = F(x)$. Instead of using an explicit Euler discretization step with fixed time increment Δt we require that the absolute value of the change of the quantity x is constant. We denote the value of this increment by $1/N$ in order to suggest that it is the weight of a numerical particle. The dynamics implies that the quantity x increases when $F(x)$ is positive and decreases if $F(x)$ is negative (in this heuristic consideration we ignore the case $F(x) = 0$). Consider therefore the scheme

$$x(t + \Delta t) = x(t) + \frac{1}{N} \text{sign}(F(x(t))),$$

where Δt has to be determined in order to ensure consistency. The above relation can be reformulated as $N|F(x(t))|(x(t + \Delta t) - x(t)) = F(x(t))$ and for consistency we need therefore $\Delta t = (N|F(x(t))|)^{-1}$. This approach does not work for systems, since the time steps required by the change of each component by the given increment are different. Note however that we do not have to change all components at once, but only sequentially. Then the selection of the components which have to be modified cannot be uniform, but according to a proper probability distribution. This feature is realized by the following method of Monte Carlo type.

Consider the multidimensional system $\dot{x} = F(x)$ with $x \in \mathbb{R}^n$ where F is a Lipschitz continuous function. The solution of this system is approximated by the Markov jump process $x(t)$ with the transitions

$$x \mapsto x + \frac{1}{N} \text{sign}(F_i(x))e_i \text{ at rate } N|F_i(x)| \quad (2)$$

where e_i denotes the i -th unit vector in \mathbb{R}^n . We change therefore the components of x sequentially and the waiting time between two consecutive jumps is exponentially distributed with parameter

$$\lambda = N \sum_{i=1}^n |F_i(x)|. \quad (3)$$

The algorithm can be then described as follows. While $t \leq t_{max}$ do:

1. Given the state $x(t)$ of the process at time t :
2. Select a component i with probability proportional to $|F_i(x(t))|$.
3. The time step $\Delta t = -\log U/\lambda$ with U uniformly distributed on $(0, 1)$ and λ given by (3) is then exponentially distributed with parameter λ .
4. Update the value of the selected component: $x_i \mapsto x_i + \frac{1}{N} \text{sign}(F_i(x))$ and set the new time as $t = t + \Delta t$.
5. GOTO 1.

We call an algorithm of this type, where the components are changed sequentially, but in a random order, a *direct ODE simulation algorithm*. This is certainly not the only possibility of approximating ordinary differential dynamics by Markov jump processes. In Section 3 we compare this approach for linear diffusion by two other methods, the random walk and the flux simulation method, where in one transition step we change two components simultaneously. It turns out that due to the fully decoupled scheme, this approach has smaller statistical fluctuations and the covariance matrix of the error process given by the central limit theorem is diagonal.

Markov jump processes can be treated either in terms of the master equations as in [7] and [11] or in terms of the martingale theory and infinitesimal generators as in [4] or [13]. The latter reference studies in fact exactly the approximation of ordinary differential equations by Markov jump processes and delivers in particular the convergence result of the direct ODE simulation method.

Indeed, we note that we have $F(x) = \sum_{x \rightarrow x'} (x' - x) R_{x \rightarrow x'}$ where the sum runs over all possible transitions from x to x' with rates $R_{x \rightarrow x'}$. According to [13] we have then

$$x_i(t) = x_i(0) + \int_0^t F_i(x(s)) ds + M_i(t) \quad (4)$$

with the martingales M_i for which holds:

$$\begin{aligned} E_x[M_i^2(t)] &= \int_0^t E_x \left[\sum_{x \rightarrow x'} ((x'_i(s) - x_i(s) + M_i(s))^2 \right. \\ &\quad \left. - M_i^2(s) - 2(x'_i(s) - x_i(s))M_i(s)) R_{x \rightarrow x'} \right] ds \\ &= \int_0^t E_x \left[\sum_{x \rightarrow x'} (x'_i(s) - x_i(s))^2 N |F_i(x(s))| \right] ds \end{aligned}$$

$$= N^{-1} \int_0^t E_x[|F_i(x(s))|] ds \quad (5)$$

where $E_x[\cdot]$ denotes conditional expectation with respect to $x(0) = x$. This result shows that the stochastic perturbation (4) of the deterministic system (without the martingale part) has magnitude $O(1/\sqrt{N})$, independent on the dimension n of the system, provided $F_i(x) = O(1)$ at all times and for all i .

Based on the theory in [13] we formulate next a result on the convergence of the algorithm.

Theorem 1. *Denote with $\varepsilon_N(t) = x_N(t) - X(t)$ the difference between a jump process $x_N(t)$ which satisfies (4) (for example defined by the transitions (2)) and the exact solution $X(t)$ of the equation $\dot{X} = F(X)$. Then for $N \rightarrow \infty$ the process $Y_N(t) = \sqrt{N} \cdot \varepsilon_N(t)$ converges weakly to the n -dimensional diffusion process $Y(t)$ given by*

$$dY(t) = \nabla F(X(t)) \cdot Y(t)dt + \sigma(X(t))dW(t) \quad (6)$$

with $Y(0) = \lim_{N \rightarrow \infty} \sqrt{N}(x_N(0) - X(0))$, provided F has uniformly continuous partial derivatives, where $W(t)$ is a n -dimensional Brownian motion and the diffusion matrix is given by $\sigma = a^{1/2}$ with

$$a_{ij}(x) = N \sum_{x \rightarrow x'} (x'_i - x_i)(x'_j - x_j) R_{x \rightarrow x'}. \quad (7)$$

For the direct ODE simulation scheme (2) we have $a_{ij}(x) = \delta_{ij} F_i(x)$, i.e. the covariance matrix $a = (a_{ij})$ is diagonal.

Proof. For a complete proof see [13], where the dynamics of the limit process is given in terms of a partial differential equation for its characteristic function. We present here only the essential steps which lead to the stochastic differential equation formulation (6). With the vector-valued martingale $M_N(t)$ from (4) we have:

$$\begin{aligned} dY_N(t) &= \sqrt{N}[F(x_N(t)) - F(X(t))]dt + \sqrt{N} \cdot dM_N(t) \\ &= \nabla F(X(t)) \cdot Y_N(t)dt + o(|Y_N(t)|)dt + \sqrt{N} \cdot dM_N(t) \end{aligned} \quad (8)$$

The desired result is obtained then by a tightness argument. The formula (7) for a is obtained by computing $E[M_{N,i} \cdot M_{N,j}]$. If the jump process is given by (2) it is easy to see that this quantity is 0 for $i \neq j$, while for $i = j$ the computation was performed in (5). \square

Corollary 1. *If we assume that $Y(0) = \lim_{N \rightarrow \infty} \sqrt{N}(x_N(0) - X(0)) = 0$, i.e. if we take for example $x_N(0) = X(0)$, then by variation of constants we obtain*

$$Y(t) = \int_0^t \Phi(t) \cdot \Phi(s)^{-1} \sigma(X(s)) dW(s) \quad (9)$$

where Φ is a fundamental system of $\dot{\Phi}(t) = \nabla F(X(t)) \cdot \Phi(t)$.

We turn now towards our case of interest, where the system of ordinary differential equations $\dot{x} = F(x)$ emerges as a spatially discrete and time-continuous version of a partial differential equation. The method presented here can be applied to partial differential equations in arbitrary space dimensions, provided the spatial discretization scheme is convergent and the solution sufficiently regular. Enhancements of this basic principle are possible, for example by considering dynamic grids in order to capture shocks of the solution. We present here this feature in the one-dimensional case (Section 6).

Given the spatially discretized system one may use standard approximation results given by the method of lines in order to estimate an error functional for the approximation of the corresponding solution of the PDE. We are interested here only in the approximation of the solution of the ODE by stochastic processes.

Under the assumption that we are dealing with a discretized PDE the matrix ∇F will exhibit a band structure. As can be seen from the previous convergence results, the stability of the deterministic scheme given by the method of lines is essential for our case too. The properties of the eigenvalues of the linearized problem are crucial in order to bound the error in (9), since if we discretize a general partial differential equation of second order, the entries of $\nabla F(x)$ will be typically of order $O(h^{-2})$. An eigenvalue with positive real part of this order of magnitude will require a very large value of N in order to reduce fluctuations and to get a good approximation of the corresponding ODE by the stochastic scheme. On the other hand, in the stable case, where the matrix norm of $\Phi(t)$ is uniformly bounded in t and h , the fluctuations are determined essentially only by σ . The behaviour of this quantity may depend on the features of the general stochastic scheme which one uses.

Consider for example the spatially discrete diffusion equation $\dot{x}_i = (\Delta_h x)_i$, $i = 1, \dots, n$, with $(\Delta_h x)_i = h^{-2}(x_{i-1} - 2x_i + x_{i+1})$, where h denotes the spatial discretization step and where we take the boundary condition $x_0 = x_{n+1} = 0$.

We then have $\Phi(t) = \exp(At)$ with

$$A = h^{-2} \begin{bmatrix} -2 & 1 & \dots & 0 \\ 1 & -2 & 1 & \vdots \\ & 1 & \ddots & 1 \\ 0 & \dots & 1 & -2 \end{bmatrix}$$

which is a typical case of stability. The other factor which influences the strength of the fluctuations is $\sigma(X(s))$, which in the case of the direct ODE simulation method equals $\text{diag}(F_i(X(s))) = \text{diag}((\Delta_h X(s))_i)$. Since X is the exact solution, in this case we can bound all terms under the integral in (9) uniformly in t and h . In Section 3 we will compare this stochastic method with two other schemes: the random walk method and flux simulation, where the entries of σ cannot be bounded uniformly in h .

Further remarks. Since the change of the components of x occurs sequentially, we may consider the value of N in equation (2) as being dependent on i and even on t : $N = N(i, t)$. This parameter can be therefore tuned and allows a more flexible implementation of the method. In the most examples considered here however it turned out that the choice of a constant N lead to the smallest statistical fluctuations of the method. An adaptive choice of N might be reasonable if an apriori analysis of the problem to be approximated suggests this approach. This is done for example in Section 6 within the framework of moving cells, where the ODE system has two types of components: cell positions and the corresponding densities. In this case the increment of the cell positions is taken much smaller than the increment corresponding to the diffusive smoothing, these two values being however constant over space and time.

From the numerical point of view it turns out to make no difference if we take deterministic time steps of length given by the expected value of the exponentially distributed random times, that is $(N \sum_i |F_i(x)|)^{-1}$. But the stochastic calculus employing martingales and generators of Markov processes requires random exponentially distributed waiting times, so we keep to this fact.

Some considerations on the computational complexity. We have to deal with two aspects: the number of operations required for updating the state of the process and the number of additional operations required by the sampling procedure and the updating of the data structures used for sampling. If we use a binary tree structure for this purpose, we need $O(\log n)$ operations for sampling and updating the binary tree, where n is the dimension of the

system. The length of a random time step between two jumps is of magnitude $N^{-1}O(n^{-1})$, therefore after N jumps we need $NO(\log n)$ operations in order to advance on a time interval of length $O(1/n)$, provided $F_i(x) = O(1)$ in all situations. We compare this result with the complexity of the explicit Euler method, for example for a stiff problem like the discretized one-dimensional heat equation. Assume that we have n equidistant grid points on $(0, 1)$. Stability requires to take time increments of $O(1/n^2)$ while in each step we perform $O(n)$ operations. In order to advance with a time interval of length $O(1/n)$ like in the previous case, we need therefore $O(n^2)$ operations. For implicit methods this amount becomes $O(n)$ (with a factor of about 8 to 10 in front). If we want to compare the efficiency, one has to note that the error of the stochastic method is given by the central limit theorem as $O(\sqrt{N}^{-1})$ and that the computational complexity is basically the same as given above for all problems. Furthermore, it makes practically no difference if we take averages over M independent realizations or if we take one realization with numerical weight $(NM)^{-1}$ instead of N^{-1} . In the very particular case of the one-dimensional diffusion equation, especially for small n , the advantage of implicit deterministic methods is evident, but with increasing n and an increase in the complexity of the problem itself the gap between general deterministic methods and the stochastic method proposed here becomes gradually smaller. Note again that the main purpose of developing the latter is to allow coupling with complex reactions which take place in every grid point and which represent in fact the bulk of the computations.

At the end of this section we mention that all simulations presented in this paper were performed in MATLAB and we specify always the real CPU-time. Since in this environment the work with dynamic data structures is more difficult like in compiled programming languages, we used instead of binary trees a stratified sampling approach, where the term $O(\log n)$ in the considerations above has to be replaced with $O(\sqrt{n})$. Nevertheless, we have to take into account also the constants of these terms (the part independent on n) which is certainly larger in the case of tree data structures. For relatively small values of n like used in our examples (between 20-80) we can say that the computational complexity is of about the same order. In any case, the examples presented here show that there is still room left to improve the efficiency by optimizing the sampling part and switching to compiled programming environments.

3. Comparison of three stochastic methods for the diffusion equation

In this section we compare the direct ODE simulation method introduced in Section 2 with two other stochastic methods: random walk and flux simulation for the one-dimensional diffusion equation $u_t = u_{xx}$ on $(0, 1)$. For this purpose we consider a uniform grid $x_i = i \cdot h$, $i = 1, \dots, n-1$, with $h = 1/n$ and denote by u_i the density in grid point x_i . At the boundary points $x_0 = 0, x_n = 1$ we consider for example Dirichlet boundary conditions: $u_0 = a, u_n = b$.

For the random walk method we consider $O(N)$ independent particles located on the grid points. We consider them to be indistinguishable and look only at the modifications of the corresponding density by the jump of a particle between two neighbouring grid points. The corresponding transitions are therefore

$$\begin{cases} u_i & \mapsto u_i - 1/N \\ u_{i\pm 1} & \mapsto u_{i\pm 1} + 1/N \end{cases} \quad \text{at rate } N \cdot h^{-2} u_i$$

which corresponds to a jump of a particle of mass $1/N$ from x_i to $x_{i\pm 1}$.

By the flux simulation method we model the transport of particles of mass $1/N$ from a cell with higher density to a neighbouring cell with lower density at rates proportional to the corresponding gradient. This scheme reads:

$$\begin{cases} u_i & \mapsto u_i + \text{sign}(\nabla_h^+ u_i)/N \\ u_{i+1} & \mapsto u_{i+1} - \text{sign}(\nabla_h^+ u_i)/N \end{cases} \quad \text{at rate } \frac{1}{2} N \cdot h^{-1} |\nabla_h^+ u_i|$$

where $\nabla_h^+ u_i := h^{-1}(u_{i+1} - u_i)$. For a pair of consecutive cells (x_i, x_{i+1}) the transport of mass takes place therefore from the cell with higher density to the cell with lower density.

Finally, for the direct ODE simulation method we have the transitions

$$u_i \mapsto u_i + \text{sign}(\Delta_h u_i)/N \quad \text{at rate } N \cdot |\Delta_h u_i|$$

where $\Delta_h u_i := h^{-2}(u_{i-1} - 2u_i + u_{i+1})$. Note that in this case we modify in a transition step only one component of the density vector, while in the previous schemes we modify always the values of two neighbouring grid points. The first two schemes are mass conservative, since mass can be lost or gained only at the boundary as specified by the boundary conditions. For the third

scheme this is not the case, the total mass being conserved only in the limit for large N .

According to the general theory in [13], all three schemes are stochastic approximations of the $n - 1$ -dimensional ODE-system $u_t = \Delta_h u$, which is ensured by the choice of the transition rates in the proper scaling: $O(Nh^{-2})$, $O(Nh^{-1})$ and $O(N)$ respectively for the random walk, flux simulation and direct ODE simulation method. This gives an indication on the speed of the three methods. Assume that the data is regularly enough, such that the values of the discrete difference operators which determine the rates are of $O(1)$. The length of the corresponding time steps implies that the direct ODE simulation is by a factor of $O(h^{-1}) = O(n)$ faster than the flux method, which in turn is about $O(n)$ times faster than the random walk method.

Concerning the approximation error, similar computations like in (5) give us that the corresponding martingale terms have second moments of order $O(N^{-1}h^{-2})$, $O(N^{-1}h^{-1})$ and $O(N^{-1})$ respectively for the random walk, flux simulation and direct ODE simulation. In this example we have $F = \Delta_h$ and we consider the CLT result given by (9). According to the comments in Section 2, the term which influences mostly the magnitude of the error is the matrix σ or the covariance matrix $a = \sigma\sigma^T$. For the random walk and the flux simulation the corresponding matrices a are tridiagonal with entries of order $O(h^{-2})$ and $O(h^{-1})$ respectively. More precisely we have $a_{ii}^{rw}(x) = h^{-2}(x_{i-1} + 2x_i + x_{i+1})$ and $a_{ij}^{rw}(x) = -h^{-2}(x_i + x_j)$ for $j = i \pm 1$ and 0 otherwise. We have further $a_{ii}^{flux}(x) = \frac{1}{2}h^{-1}(|\nabla_h^+ x_{i-1}| + |\nabla_h^+ x_i|)$ and $a_{ij}^{flux}(x) = -\frac{1}{2}h^{-1}|\nabla_h^+ x_j|$ for $j = i \pm 1$ and 0 otherwise. For the direct ODE simulation method the matrix is diagonal and we have $a_{ii}^{dODE}(x) = F_i(x) = (\Delta_h x)_i = h^{-2}(x_{i-1} - 2x_i + x_{i+1})$. These considerations show that for sufficiently regular data, the direct ODE simulation method is superior also concerning accuracy to the other two.

The above remarks are also backed up by the numerical experiments as shown in Figure 1. As initial condition we take $u_0(x) = x^2$ and the boundary conditions are $u(t, 0) = 0, u(t, 1) = 1$.

The corresponding CPU times for the simulations of 10 independent realizations were of 0.56 sec. (direct ODE simulation), 30.50 sec. (flux method) and 612.50 sec. (random walk method). This picture illustrates the previous theoretical considerations concerning speed and stochastic fluctuations. The advantage of the direct ODE simulation method is enhanced also by the fact that in this case we have $|\Delta_h u| \ll |\nabla_h^+ u|, u$.

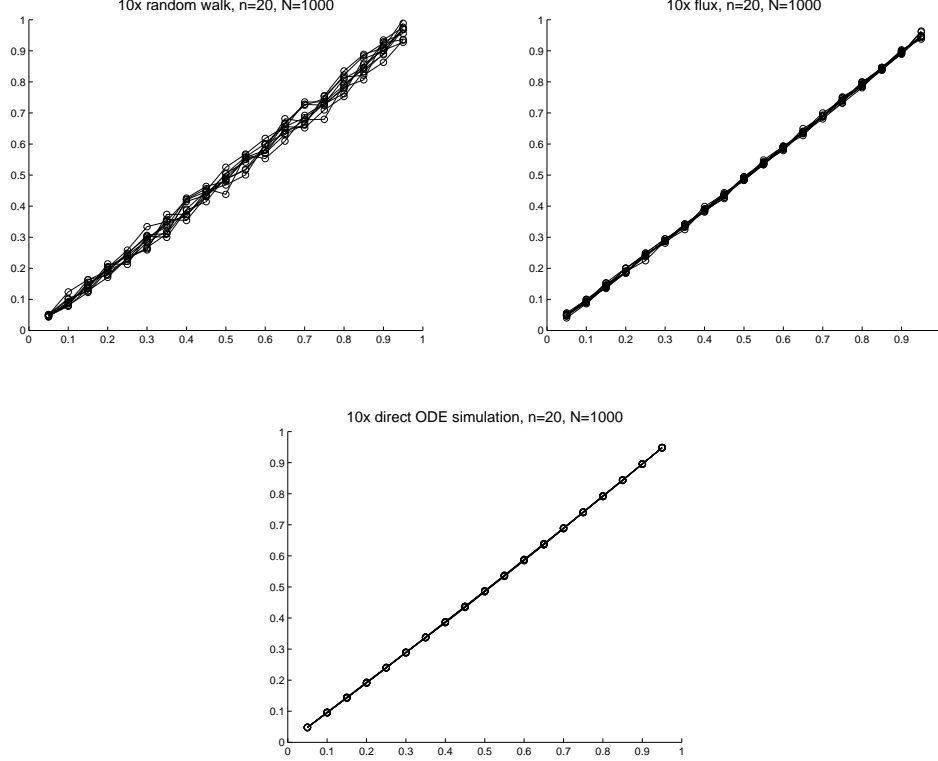


Figure 1: Comparing 10 realizations of the three methods at time $t = 0.3$, $u_0(x) = x^2$.

We consider next a situation where we have (at least locally) $|\Delta_h u| \gg |\nabla_h^+ u|, u$. This is the case for the initial condition $u_0(x) = 6x(1-x)$ and zero Dirichlet boundary conditions. The results of this simulation are presented in Figure 2.

The CPU times for the simulation of 10 independent realizations were of 2.59 sec. (direct ODE simulation), 25.87 sec. (flux method) and 315.79 sec. (random walk method). Note that the variance of the computations produced by the flux method is smaller in the middle of the interval, where the derivative of the solution (and therefore the local rate) is close to 0. Even with the factor h^{-1} in front, it is smaller than the variance of the direct ODE simulation, since there the absolute value of the second derivative is maximal. But in the points where the first derivative of the solution stays away from 0, the variance of the direct ODE simulation method is in turn smaller. This behaviour indicates that locally the flux scheme might be an alternative for

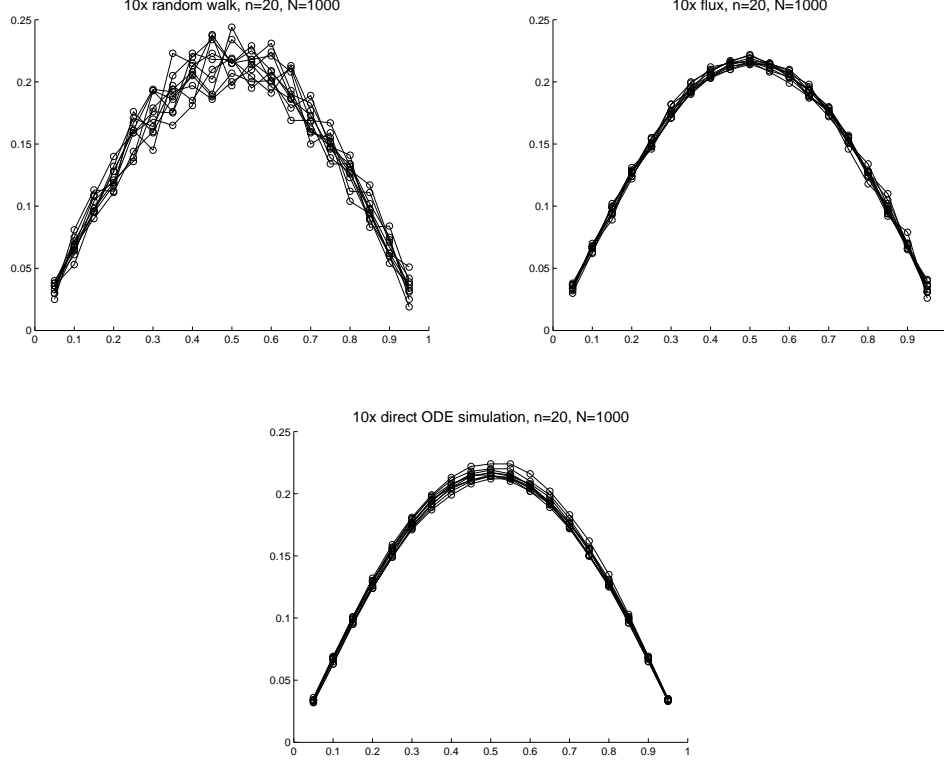


Figure 2: Comparing 10 realizations of the three methods at time $t = 0.2$, $u_0(x) = 6x(1 - x)$.

the direct ODE simulation method if one realizes a coupling of both types of the dynamics. But one has also to take into account the execution times. In most cases the use of an adaptive grid or of a larger value of N for the direct ODE simulation method will turn out to be more efficient.

We will take next a closer look at the dynamics induced by the three schemes. For bounded, continuous test functions Φ the general theory of Markov processes ([4]) gives us the martingale representation

$$\Phi(u(t)) = \Phi(u(0)) + \int_0^t \Lambda \Phi(u(s)) ds + M_\Phi(t)$$

with the martingale term $M_\Phi(t)$ and where Λ is the infinitesimal generator of the Markov process $u(t)$.

In our special case we take $\Phi(u) = \langle u, \phi \rangle$, i.e. the euclidean scalar prod-

uct with a given $n - 1$ -dimensional vector ϕ . We obtain then the following dynamics:

1. for the random walk method:

$$\langle u(t), \phi \rangle = \langle u(0), \phi \rangle + \int_0^t \langle u(s), \Delta_h \phi \rangle ds + M_\phi^{rw}(t)$$

2. for the flux simulation method:

$$\langle u(t), \phi \rangle = \langle u(0), \phi \rangle - \int_0^t \langle \nabla_h^+ u(s), \nabla_h^+ \phi \rangle ds + M_\phi^{flux}(t)$$

3. for the direct ODE simulation method:

$$\langle u(t), \phi \rangle = \langle u(0), \phi \rangle + \int_0^t \langle \Delta_h u(s), \phi \rangle ds + M_\phi^{dODE}(t).$$

By considering summation by parts in the case of zero Dirichlet or Neumann boundary conditions we obtain that the scalar products within the three integral terms above are in fact equal. Nevertheless, the form presented here points out towards the weak formulation of the dynamics, where one derivative is applied on the test function (flux simulation) or even two derivatives are applied on the test function (random walk). This observation together with the previous remarks on speed and precision leads to the following conclusion: the regularity of the data determines which type of method one has to chose for approximating difference quotients of second order in general problems. Regular solutions suggests the direct ODE simulation method, while lack of regularity may lead to the choice of a flux-scheme which approximates some weak formulations of the problem. Random-walk-type methods are suited only if even less regularity is present.

In this example we can speak of models which take place naturally on three different levels of spatial resolution (if we want to approximate the deterministic equation by the same level of precision). The random walk method gives us the right dynamics on a very fine scale, where a numerical particle almost simulates a “physical particle”. On a coarser scale (in space and time) we may consider not individual jumps of numerical particles, but the net difference of jumps in both directions, i.e. the flux. The numerical particle in the flux simulation method with weight $1/N$ lives therefore in a different world. Finally, if the regularity of the data allows this, we may

switch to an even coarser scale in space and time, where we consider only the individual changes in the density value at each grid point due to the cumulated dynamics corresponding to the particle jumps or to the flux. By this successive transitions among different scales we remain within the same level of precision. However, if we compare the three methods on the same resolution level, which is the relevant situation in practical applications, we see in general the advantage of the direct ODE simulation method over the other two.

4. Reaction-diffusion problems

In this section we consider an application of the direct ODE simulation method to reaction-diffusion problems. The first problem we consider is

$$\frac{du}{dt} = \Delta u + \frac{5e^\delta}{\delta}(2 - u) \exp(-\delta/u)$$

with boundary conditions: $\partial_\nu u(0) = 0$, $u(1) = 1$ and initial condition $u_0 \equiv 1$.

This equation models a problem in combustion theory (see [6] and [11], p.439). u denotes a temperature which increases gradually up to a critical value when ignition occurs, resulting in a fast propagation of a reaction front towards the right end of the interval. It is a typical benchmark problem for deterministic methods which use adaptive grids (in time and space).

In our case we consider a uniform spatial grid, while the time adaptivity is given intrinsically by the stochastic scheme. Note that we consider the values F_i in the transition rates of the stochastic process to be the fully discretized r.h.s. of the reaction-diffusion equations above. That is, we do not split the dynamics in reaction part and diffusion part. This choice ensures an almost two times faster execution at the same precision, since the values $|F_i|$ in this case are smaller than in a splitting approach, due to the partial cancelling effects of reaction and diffusion. The results are plotted in Figure 3 and are comparable to those in [6].

The maximal error for our method, as well as that of the deterministic methods in [6] is around the ignition time computed as $t = 0.26$. But while a deterministic numerical scheme with an induced error gives no hint on the position of the exact solution, we may now take advantage on the statistical properties of the Monte Carlo -type method and conclude that the exact solution is close to the average of the curves computed in several independent runs. That is, about in the middle of the 'stripes' produced by the stochastic

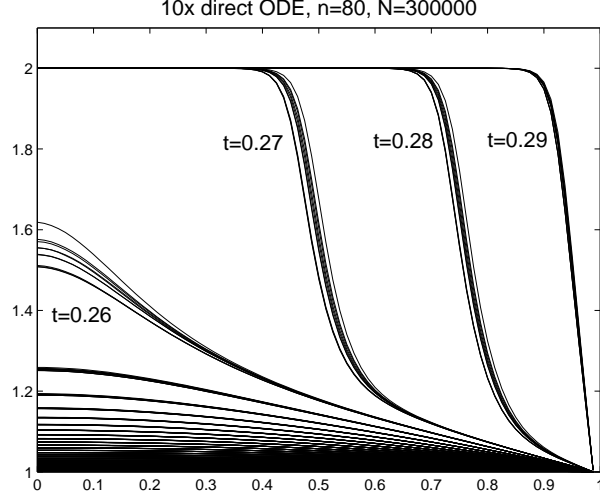


Figure 3: $\delta = 20$, uniform grid, 10 realizations, CPU-time ≈ 9 min/run

fluctuations of the scheme. In order to have a good time resolution of the fast motion of the front which takes place within the small time interval $[0.26, 0.29]$ we had to consider the very large value of $N = 300\,000$.

In the next section we will show also the possibility of approximating diffusion on non-uniform grids. But in this example a naive refinement of the spatial grid in the region of the front leads to no improvement, due to the larger statistical fluctuations which are induced. However, it is possible to use moving grid techniques like in [6] in order to follow the motion of a grid which is always fully adapted to the traveling front. We do not follow this path here, but in Section 6 we present the use of moving grids for convection-diffusion problems, where the grid is moving with the natural velocity given by the problem.

The next example is an application to the Fisher-Kolmogorov equation

$$\frac{du}{dt} = D\Delta u + u - u^2$$

with $u(0, x) = 1$ for $x \leq 0$ and $u(x, 0) = 0$ for $x \geq x_1$, which in [3] is simulated by a stochastic scheme based on the random walk and where diffusion and reaction were decoupled. The solution of this equation exhibits a traveling front, a fact that can be seen in Figure 4 which is based on simulations with the same parameters as in [3], adapted to our setting.

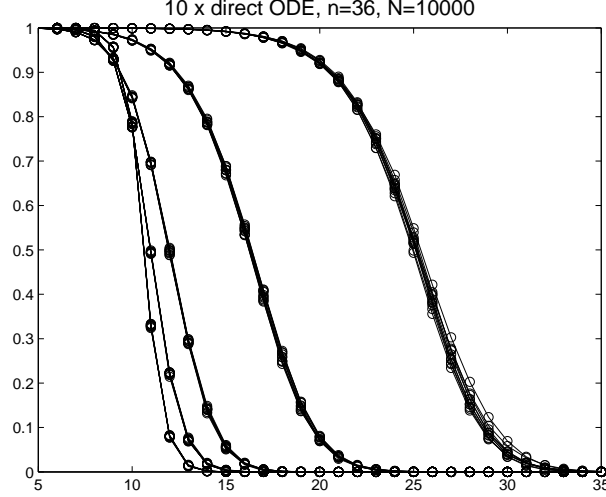


Figure 4: 10 realizations, solution at times $t = 0.5, 1, 2, 5, 10$, CPU-time ≈ 2.8 sec/run

Note that in the algorithm presented in [3] there exists only one scaling parameter, which is the spatial discretization step h in our notation. The densities are taken as the number of particles divided by the volume of the cells, that is by h , which would correspond to the choice $N = h^{-1}$ in our situation. Due to the large fluctuations induced by the random walk method the asymptotic speed of the traveling waves was computed in [3] (especially for relatively large h) to be in the average different from the theoretical predicted value. Note that this scaling is very particular and it approximates the limiting partial differential equation for $h \rightarrow 0$, but for relatively large h it can be far away from the deterministic ODE which corresponds to the spatially discretized PDE. Our experiments by the direct ODE simulation method do not exhibit such a behaviour, which can be explained by the relatively small fluctuations of our scheme.

5. Diffusion on nonuniform grids

In contrast to explicit deterministic schemes, due to its natural time-adaptivity, the direct ODE simulation method turns out to be stable also on non-uniform grids. Consider a grid $x_0 = a < x_1 < x_2 < \dots < x_{n-1} < x_n = b$ with $h_i = x_i - x_{i-1}$ and on it the following transitions:

$$u_i \mapsto u_i + \frac{\text{sign}(\Delta_h u_i)}{N} \text{ at rate } N \cdot |\Delta_h u_i|,$$

for $i = 1, \dots, n-1$, where $\Delta_h u_i = \alpha_i u_{i-1} + \beta_i u_i + \gamma_i u_{i+1}$, with

$$\alpha_i = \frac{2}{h_i(h_i + h_{i+1})}, \quad \gamma_i = \frac{2}{h_{i+1}(h_i + h_{i+1})}, \quad \beta_i = -\alpha_i - \gamma_i.$$

These are the values required for the consistency of the space discretization scheme. The values u_0 and u_n at the boundary points are prescribed by the boundary conditions.

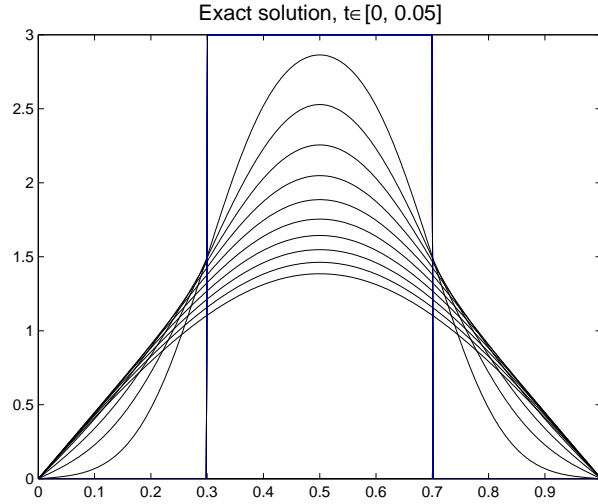


Figure 5: Exact solution at equidistant time steps $t \in [0, 0.05]$.

We present an example for the diffusion equation with discontinuous initial data. A plot of the exact solution at equidistant time steps is given in Figure 5.

We compare the deterministic explicit and implicit (Crank-Nicolson) finite difference schemes on uniform grids with the direct ODE simulation method on a nonuniform grid. This is generated very naively, only for illustration purposes, as a coarse grid on which we superpose locally two finer grids around the discontinuity points of the initial data.

As shown in Figure 6 the error is about the same, but the number of grid points used by the direct ODE simulation method is of about three times

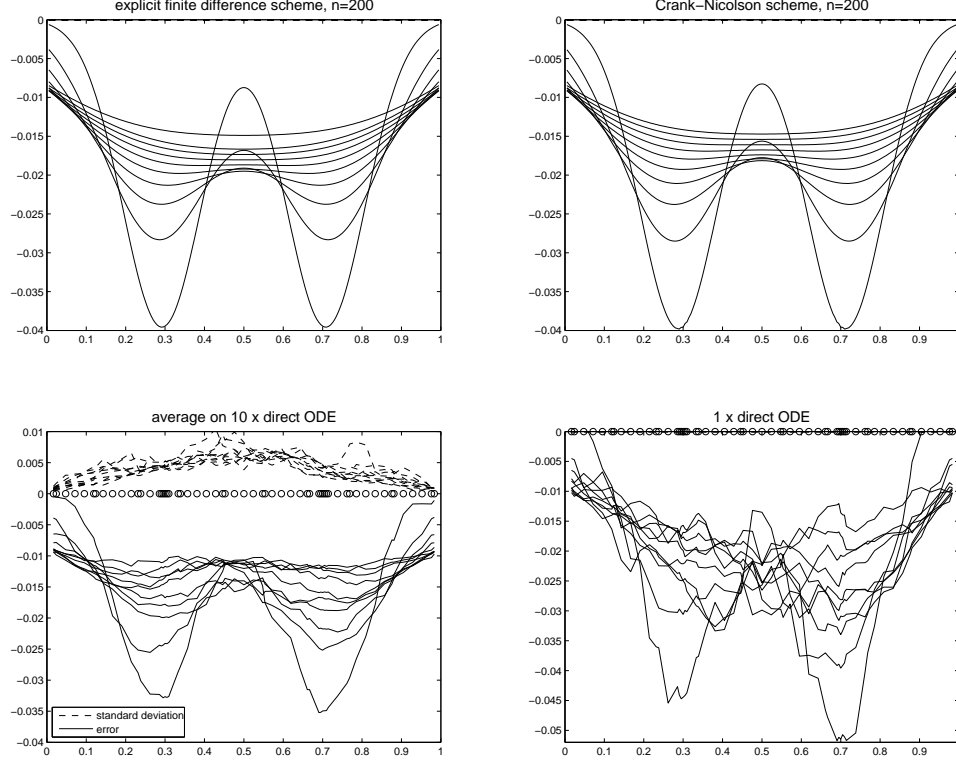


Figure 6: Difference to the exact solution at equidistant time steps $t \in [0, 0.05]$.

smaller. The CPU times are of 0.15 sec. (Crank-Nicolson with MATLAB routines for matrix operations), 0.59 sec. (explicit finite difference) (both with 200 grid points) and 0.82 sec. for one realization of the direct ODE simulation method based on 60 grid points with $N = 1000$. In the plot of the average taken on 10 realizations the standard deviation of each curve is also represented (the dashed line). The fluctuations are small and the error of one typical realization of the method is comparable to that of the deterministic schemes. We note the relatively large execution time of the direct ODE simulation method compared to the results in Section 3. The explanation lies in the discontinuity of the initial data and in the large values of the discrete difference operators at small times. This implies extremely large rates and consequently short time steps and more iterations. This example shows also the effect of the automatic time adaptivity of the scheme presented here.

The possibility of using this scheme for approximating diffusion on nonuniform grids will be exploited in the next section in the context of convection-diffusion equations.

6. Convection-diffusion dynamics with moving cells

For the viscous Burgers' equation

$$\frac{du}{dt} + uu_x = \nu u_{xx} \quad (10)$$

we consider the following variant of the direct ODE simulation method. For $i = 1, \dots, n$ let $x_i(t)$ denote the position of cell i and $u_i(t)$ the density in cell i at time t . We consider the ODE system

$$\begin{cases} \dot{x}_i = u_i \\ \dot{u}_i = \nu \Delta_h u_i \end{cases}$$

with Δ_h defined as in Section 5 on the dynamic grid $x_i = x_i(t)$. This means that the cells with positions x_i and densities u_i follow a Lagrangian dynamics with a velocity which equals u_i , while on the non-uniform grid configuration we simulate diffusion by the known scheme. This system is approximated by the direct ODE simulation method, where we have to deal with some computational issues:

- We take $N_x = 10 \cdot N_u$, that is the increment by which we change the position of the cells is 10 times smaller than the increment by which we change the density values by diffusion.
- In the numerical simulations the trajectories of the cells may cross, or may become very close. In this case we eliminate one of the cells, in order to avoid the numerical explosion of the diffusion rates.
- Due to several elimination steps the number of cells can decay significantly. A typical rule used in this situation is the following: if the current number of cells becomes e.g. half of the initial number, we add the missing number of cells by interpolation, by keeping at the same time the remaining cells. A completely new cell configuration would induce additional errors, since the motion was simulated in a natural way (and with a higher resolution than diffusion), so the position of the already existing cells may be assumed to be very precise.

- Depending on the boundary conditions, we have to consider a possible inflow or outflow of cells at the boundaries of the given computational domain.

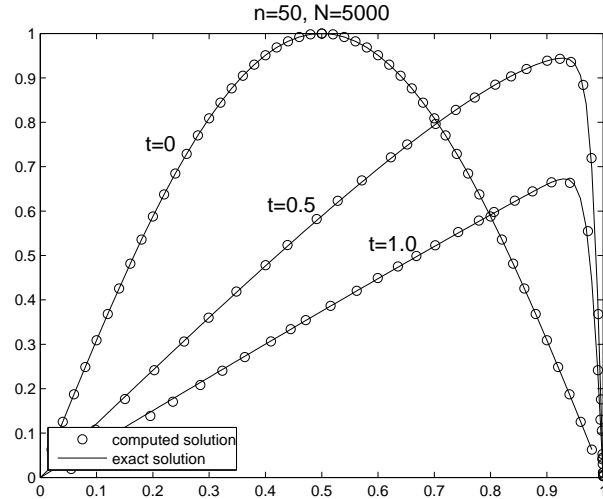


Figure 7: Comparison with the exact solution for $\nu = 10^{-2}$. CPU-time ≈ 1 min.

The first example is to approximate the equation (10) on the interval $(0, 1)$ with zero boundary conditions and initial condition $u_0(x) = \sin \pi x$, see also [5]. In this case we can compare the numerical results with the exact solution as shown in Figure 7.

The second example we consider is the problem of two moving fronts. The domain of definition is now given by the whole real line and an explicit solution can be computed as a combination of exponential functions, see [12], p.164 and p.201. The upper front travels with a higher velocity and catches up the slower one. The two fronts then merge and their common motion continues as one single front.

In Figure 8 we show the results of the numerical simulations produced by our method. Note that in this case we have not only a natural time-adaptivity, but also a natural grid-adaptivity. The value $N = 5000$ regards the parameter of the diffusion dynamics. As already mentioned, the corresponding parameter for the convective dynamics is taken as $10N$ in order to ensure a better spatial resolution. The same (larger) value for the diffusion dynamics would lead to a significant increase in the execution time.

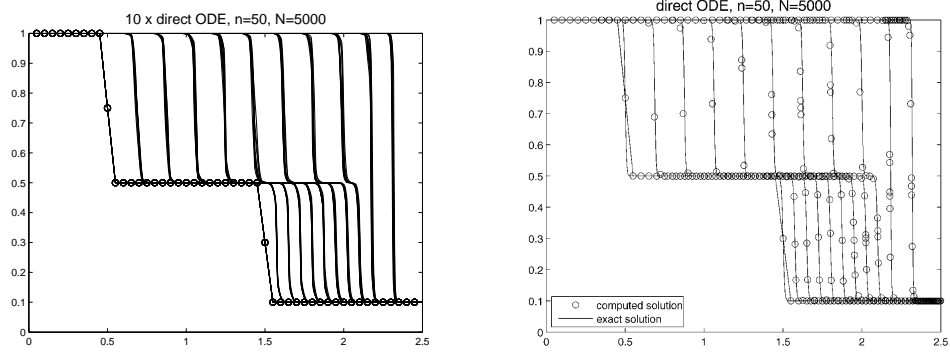


Figure 8: Computations for $\nu = 10^{-3}$ and $t \leq 2.5$. 10 realizations (left) and one realization compared to the exact solution (right). CPU-time $\approx 1\text{min/run}$

7. A free boundary problem: the Black-Scholes equation for American options

We consider the Black-Scholes equation for American put-options (formulated forward in time instead of backward):

$$P_t = \frac{1}{2}\sigma^2 S^2 P_{SS} + r(P_S - P) \text{ for } S > S_f,$$

$$P(S, t) = K - S \text{ for } S \leq S_f(t).$$

with the free boundary S_f and the initial conditions:

$$P(S, 0) = (K - S)^+, \quad S_f(0) = K.$$

$P(S, t)$ is the value of the option at time t if the underlying asset has the value S . The free boundary point S_f is the critical value of the underlying asset such that for $S \leq S_f$ it is profitable to exercise the option, its value being given by the payoff $K - S$. For $S > S_f$ it is more profitable to hold the option, the value being given then by the solution of the Black-Scholes equation.

Usual deterministic schemes for this equation are based on its formulation as an obstacle problem and employing for example projection SOR (successive over-relaxation) methods for the numerical approximation of the corresponding differential inequalities. In contrast to this approach, the application of the direct ODE simulation method is straightforward. Due to

the dynamics of the problem, the free boundary S_f will move to the left. Consider a discretized approximation of the initial value on a uniform grid and denote by i^* the index of the grid point corresponding to the free boundary. We simulate the dynamics of the Black-Scholes equation only on the interval (S_{i^*}, S_n) which, at a certain moment, will imply also a change of the value S_{i^*} . After this step we move the boundary one point to the left, that is we set $i^* = i^* - 1$ and for the new value of $P(S_{i^*})$ we take the value computed in the initial condition, that is $K - S_{i^*}$. Up to this modification, the algorithm follows the usual pattern. Note that we simulate the motion of the free boundary in a natural way, as a result of the infinitesimal dynamics of the problem. The numerical results for the parameter values $K = 100, r = 0.03, \sigma = 0.4, T = 1$ are given in Figure 9.

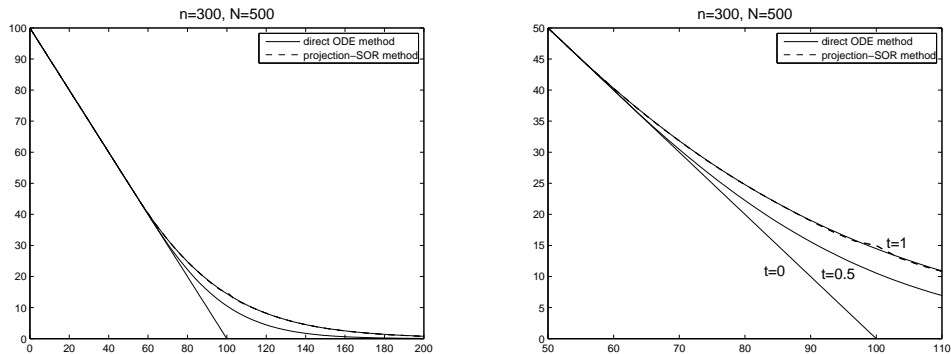


Figure 9: Value of an American put option computed by two methods at times $t = 0, 0.5, 1$. CPU-time ≈ 6 sec (for the direct ODE simulation).

We can see a good agreement of the results produced by the direct ODE simulation method (one realization) and a standard deterministic scheme. The latter runs in a time of under 1 second, but it uses the fast MATLAB-routines for matrix operations, while the samplings and the explicit updates of the stochastic process are performed always “by hand”.

8. Conclusions

In this paper we presented a stochastic scheme for approximating solutions of ordinary differential equations by direct simulation of the corresponding infinitesimal dynamics. The method is explicit and it can be applied

also for semi-discrete partial differential equations, even on non-uniform spatial grids (one-dimensional). Time-adaptivity is provided intrinsically, while space adaptivity can be also employed. An error analysis shows that the fluctuations are small, provided the deterministic solution which has to be approximated stays bounded on the given time interval. Given the dimension n of the system and the parameter N which induces an approximation error of $O(\sqrt{N}^{-1})$, the scheme requires $NO(\log n)$ operations in order to advance on a time interval of length $O(1/n)$. Comparing with deterministic schemes for systems of dimension n , which require a linear or quadratic complexity in n for the same time interval, we can say that the method proposed here could be more efficient for large systems, for example in 2 or 3 space dimensions and where we have a large number of species which interact in every cell or grid point. A typical application example can be a spatially inhomogeneous coagulating system, where in each cell one has to simulate an infinite-dimensional system of ordinary differential equations.

The general principle of the direct ODE simulation method can be applied for arbitrary evolution partial differential equations, but in the present paper we show applications only at convection-diffusion-reaction problems in one space dimension, including free boundary problems. Further research towards improving this method can be directed at variance reduction and generalizations in more space dimensions in the case of non-uniform grids, for example in a finite-element setting.

References

- [1] D. Aldous, Deterministic and stochastic models for coalescence (aggregation and coagulation): a review of the mean-field theory for probabilists, *Bernoulli* 5 No. 1 (1999) 3-48
- [2] L. Arnold, M. Theodosopulu, Deterministic limit of the stochastic model of chemical reactions with diffusion, *Adv.Appl.Prob.* 12 (1980) 367-379
- [3] H.-P. Breuer, W. Huber, F. Petruccione, Fluctuation effects on wave propagation in a reaction-diffusion process, *Physica D* 73, No.3 (1994) 259-273
- [4] S. Ethier, T.G. Kurtz, *Markov Processes: Characterization and Convergence*, Wiley, 1986

- [5] C.A.J. Fletcher, A comparison of finite element and finite difference solutions of the one- and two-dimensional Burgers' equations. *J.Comput.Phys.* 51 (1983), 159-188
- [6] R. M. Furzeland, J.G. Verwer, P.A. Zegeling, A numerical study of three moving-grid methods for one-dimensional partial differential equations which are based on the method of lines, *J.Comput.Phys.* 89 (1990), 349-388
- [7] D.T. Gillespie, *Markov Processes*, Academic Press, San Diego, 1992
- [8] F. Guiaş, Convergence properties of a stochastic model for coagulation-fragmentation processes with diffusion, *Stochastic Anal.Appl.* 19 No.2 (2001) 245-278
- [9] F. Guiaş, A stochastic numerical method for diffusion equations and applications to spatially inhomogeneous coagulation processes, in: H. Niederreiter, D. Talay (Eds.). *Monte Carlo and quasi Monte Carlo methods 2004*, Springer, 2006, pp.147-162
- [10] F. Guiaş, A stochastic approach for simulating spatially inhomogeneous coagulation dynamics in the gelation regime, *Comm. Nonlinear Sci. Numer. Simulat.* 14 No.1 (2009), 204-222, doi:10.1016/j.cnsns.2007.07.015
- [11] J. Honerkamp, *Stochastic Dynamical Systems*, VCH, New-York, 1994
- [12] W. Hundsdorfer, J.G. Verwer, *Numerical Solution of Time-Dependent Advection-Diffusion-Reaction Equations*, Springer, Berlin-Heidelberg, 2003
- [13] T.G. Kurtz, Limit theorems for sequences of jump Markov processes approximating ordinary differential processes, *J.Appl.Prob.* 8 (1971) 344-356
- [14] B. Lapeyre, É. Pardoux, R. Sentis, *Introduction to Monte-Carlo Methods for Transport and Diffusion Equations*, Oxford University Press, 2003

Preprints ab 2008

- 2009-04 **Flavius Guias**
Direct simulation of the infinitesimal dynamics of semi-discrete approximations for convection-diffusion-reaction problems
- 2009-03 **Franz Kalhoff and Victor Pambuccian**
Existential definability of parallelism in terms of betweenness in Archimedean ordered affine geometry
- 2009-02 **Fulvia Buzzi, Michael Lenzinger and Ben Schweizer**
Interface conditions for degenerate two-phase flow equations in one space dimension
- 2009-01 **Henryk Zähle**
Approximation of SDEs by population-size-dependent Galton-Watson processes
- 2008-25 **Winfried Hazod**
Mehler semigroups, Ornstein-Uhlenbeck processes and background driving Lévy processes on locally compact groups and on hypergroups
- 2008-24 **Karl Friedrich Siburg, Pavel A. Stoimenov**
Symmetry of functions and exchangeability of random variables
- 2008-23 **Ina Kirsten Voigt**
Voronoi Cells of Discrete Point Sets
- 2008-22 **Michael Lenzinger and Ben Schweizer**
Effective reaction rates of a thin catalyst layer
- 2008-21 **Michael Voit**
Bessel convolutions on matrix cones: Algebraic properties and random walks
- 2008-20 **Margit Rösler and Michael Voit**
Limit theorems for radial random walks on $p \times q$ -matrices as p tends to infinity
- 2008-19 **Michael Voit**
Central Limit Theorems for Radial Random Walks on $p \times q$ Matrices for $p \rightarrow \infty$
- 2008-18 **Michael Voit**
Limit theorems for radial random walks on homogeneous spaces with growing dimensions
- 2008-17 **Ansgar Steland and Henryk Zähle**
Sampling inspection by variables: nonparametric setting
- 2008-16 **Guy Bouchitté and Ben Schweizer**
Homogenization of Maxwell's equations with split rings

- 2008-15 **Wilfried Hazod**
Multiple selfdecomposable laws on vector spaces and on groups:
The existence of background driving processes
- 2008-14 **Wilfried Hazod**
Mixing of generating functionals and applications to (semi-)stability
of probabilities on groups
- 2008-13 **Wilfried Hazod**
Probability on Matrix-Cone Hypergroups: Limit Theorems and
Structural Properties
- 2008-12 **Michael Lenzinger and Ben Schweizer**
Two-phase flow equations with outflow boundary conditions
in the hydrophobic-hydrophilic case
- 2008-11 **Karl Friedrich Siburg**
Geometric proofs of the two-dimensional Borsuk-Ulam theorem
- 2008-10 **Peter Becker-Kern, Wilfried Hazod**
Mehler hemigroups and embedding of discrete skew convolution
- 2008-09 **Karl Friedrich Siburg, Pavel A. Stoimenov**
Gluing copulas
- 2008-08 **Karl Friedrich Siburg, Pavel A. Stoimenov**
A measure of mutual complete dependence
- 2008-07 **Karl Friedrich Siburg, Pavel A. Stoimenov**
A scalar product for copulas
- 2008-06 **Flavius Guias**
Generalized Becker-Döring Equations Modeling the Time Evolution of
a Process of Preferential Attachment with Fitness
- 2008-05 **Benjamin Fine, Alexei Myasnikov, Gerhard Rosenberger**
Generic Subgroups of Group Amalgams
- 2008-04 **Ben Schweizer**
Homogenization of the Prager model in one-dimensional plasticity
- 2008-03 **Benjamin Fine, Miriam Hahn, Alexander Hulpke, Volkmar
große Rebel, Gerhard Rosenberger, Martin Scheer**
All Finite Generalized Tetrahedron Groups
- 2008-02 **Benjamin Fine, Gerhard Rosenberger**
An Epic Drama: The Development of the Prime Number Theorem
- 2008-01 **Henryk Zähle**
Weak approximation of SDEs by discrete-time processes